

REMARKS

Petition for Extension of Time Under 37 CFR 1.136(a)

It is hereby requested that the term to respond to the Examiner's Action of July 17, 2008 be extended two months, from October 17, 2008 to December 17, 2008.

The Commissioner is hereby authorized to charge the extension fee to Despoit Account No. 50-4364 and any additional fees associated with this communication to Deposit Account No. 09-0461.

In the Office Action, the Examiner indicated that Claims 1 through 20 are pending in the application and the Examiner rejected all claims.

The §101 Rejection

On page 3 of the Office Action, the Examiner has rejected claims 8-14 under 35 U.S.C. § 101 as being directed to non-statutory subject matter. Per the Examiner's suggestion, applicant has amended both the specification (page 9) and claim 8 to specifically recite that the medium on which software code resides is a storage medium, as is supported elsewhere in the specification. Accordingly, the Examiner is respectfully requested to reconsider and withdraw the rejection of claims 8-14 under 35 U.S.C. §101.

Claim Rejections, 35 U.S.C. §102 and §103

On page 4 of the Office Action, the Examiner rejected Claims 1-2, 4-9, 11-16, and 18-20 under 35 U.S.C. §102(e) as being anticipated by "Secure Java Class Loading", IEEE Internet

Computing, November/December 1998, Pages 56-61 by Gong (“Gong”). On page 10 of the Office Action, the Examiner rejected Claims 3, 10 and 17 under 35 U.S.C. §103(a) as being unpatentable over Gong in view of U.S. Application Publication No. 2003/0115218 to Bobbitt et al. (“Bobbitt”).

The Present Invention

The present invention is a method, system, and computer program product for compiling Java code. In accordance with the present invention, Java code that references classes residing in a workspace can be compiled. In accordance with the present invention, a workspace identifier is placed within the classpath to indicate the location of the referenced classes that may reside within a workspace. Specifically, Claim 1 recites: “1) determining if a referenced class file is located in a workspace; 2) locating the class file; 3) accessing the class file; and 4) returning the class file data to the compiler wherein said compiler reads said class data file to produce machine executable code without removing any class data files from said workspace.”(lines 3-8) In accordance with a preferred embodiment, the files on a web site are serviced using a file database, and a class file is allocated to a workspace by creating an additional entry in the file database. The class file is invoked by the compiler through the database and processed without moving the location of the class file in the workspace.

“Secure Java Class Loading”, IEEE Internet Computing, Nov/Dec 1998, Pages 56-61 by Gong

“Secure Java Class Loading”, IEEE Internet Computing, November/December 1998, Pages 56-61 by Gong (“Gong”) teaches dynamic class loading as a feature of the Java virtual machine.

Gong defines several unique characteristics of class loading, including loading classes on demand, on a just-in-time basis. Gong teaches applying Java ClassLoaders to execute Java byte-codes (applications, applets, etc.). Java byte-codes are executed using a JVM which includes ClassLoaders as a feature.

The Cited Prior Art Does Not Anticipate the Claimed Invention

The MPEP and case law provide the following definition of anticipation for the purposes of 35 U.S.C. §102:

“A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference.” MPEP §2131 citing *Verdegaal Bros. v. Union Oil Company of California*, 814 F.2d 628, 631, 2 U.S.P.Q. 2d 1051, 1053 (Fed. Cir. 1987)

The Examiner Has Not Established a *Prima Facie* Case of Anticipation

The Examiner is applying the Java ClassLoader methods of Gong to the present invention. This is not warranted and not taught or suggested by Gong. The claimed invention involves compiling Java source files, and not executing Java byte-codes as is done by Gong. Java ClassLoaders are used in executing Java byte-codes, as in Gong. There is no similar concept in compilers that compile Java source files. Java byte-codes are executed using a JVM which includes ClassLoaders as a feature. Compiling Java source files is done by a compiler, i.e., there is no ClassLoader concept in compilers.

Gong is describing ClassLoaders as used when executing Java byte-codes (applications, applets, etc). There is no discussion about running a compiler using class data files as claimed in the

present invention. Each of the independent claims has been amended to point out that the compiler performs a "read" process as opposed to an "execute" process, in an attempt to further distinguish the claims from the teachings of Gong, which relate to executing Java byte-codes.

One of ordinary skill in the art would not apply ClassLoader concepts in general to a Java compiler compiling source code. Applicant knows of no compiler that does this, and Gong does not teach or suggest it. Compiling of source files is a completely independent discipline from executing byte-codes. Applicant's claimed invention is not a ClassLoader (as in Gong), nor does it work like one.

Each of the independent claims (and thus all of the claims) recite the novel aspect described above, that is, running a compiler using class data files. Accordingly, all of the claims patentably define over Gong and are in condition for allowance. The Examiner is respectfully requested to reconsider and withdraw the rejection of the claims under 35 U.S.C. §102.

The Examiner Has Not Established a *Prima Facie* Case of Obviousness

Bobbit, likewise, neither teaches nor suggests running a compiler using class data files. Accordingly, none of the claims are rendered obvious by a combination of Gong and Bobbit. Accordingly, the Examiner is respectfully requested to reconsider and withdraw the rejections of Claims 3, 10, and 17 under 35 U.S.C. §103.

Conclusion

The present invention is not taught or suggested by the prior art. Accordingly, the Examiner is respectfully requested to reconsider and withdraw the rejection of the claims. An early Notice of Allowance is earnestly solicited.

The Commissioner is hereby authorized to charge the extension fee to Deposit Account No. 50-4364 and any additional fees associated with this communication to Deposit Account No. 09-0461.

Respectfully submitted,

December 17, 2008
Date

/Mark D. Simpson/
Mark D. Simpson, Esq.
Registration No. 32,942

SAUL EWING LLP
Centre Square West
1500 Market Street, 38th Floor
Philadelphia, PA 19102-2189
Telephone: 215 972 7880
Facsimile: 215 972 4169
Email: MSimpson@saul.com